7090/7094 DATA PROCESSING SYSTEMS BULLETIN

IBM 7090 PROGRAMMING SYSTEMS
SHARE 7090 9PAC
SUPPLEMENT

This edition is a major revision of Form J28-6211 and obsoletes the previous edition. Two new sections are included: (1) Addenda and Errata to Part 4, (2) File Processor with Reports Generator Capabilities.

Other factual changes or additions in this edition are marked with a dot (●).

Each page on which a change or addition appears has a dot (●) in the upper outside corner.

This publication contains additions and corrections to the following bulletins which make up the 9PAC Reference Manual:

SHARE 7090 9PAC
Part 1: Introduction and General Principles
Form J28-6166

SHARE 7090 9PAC
Part 2: The File Processor
Form J28-6167

SHARE 7090 9PAC
Part 3: The Reports Generator
Form J28-6168

SHARE 7090 9PAC
Part 4: Operating Instructions
Form J28-6215

This publication describes: (1) the preparation of 9PAC files for use with the fixed length version of 7090/7094 Sort, described in the IBM bulletin, IBM 7090/7094 Generalized Sorting Program: 7090/7094 Sort (729 - Fixed Length), Form J28-6217 and (2) the File Processor with Reports Generator Capabilities.

References to the I-string report output feature in the bulletins listed above should be disregarded since this feature will not be implemented.

Address comments regarding this publication to
Programming Systems Publications, IBM Corporation, P. O. Box 390, Poughkeepsie, N. Y.

TABLE OF CONTENTS

ADDENDA AND ERRATA TO PART 1

The following addenda and corrections should be made on the pages listed.

p. 1 Add as paragraph 4 of "An Introduction to 9PAC," the following:

Sorting is a distinct function used in data processing of business transactions. 9PAC files may be sorted with the 7090/7094 Sort program which operates under the Basic Monitor (IBSYS). This sort program is described in the IBM bulletin, IBM 7090/7094 Generalized Sorting Program: 7090/7094 Sort (729 - Fixed Length), form J28-6217.

p.24 Replace the entry for column 12 with:

Open File Indicator — Must contain one of the following codes to indicate action to be taken when the file is opened.

R or b	Rewind before opening.
N	No rewind.
S	No rewind, no label action.

● p.25 Replace the entries for columns 18-21 with:

The unit assignment routines used by 7090 9PAC are identical with those used by 709/7090 IOCS. For information concerning these procedures, refer to the IBM reference manual, IBM 709/7090 Input/Output Control System, form C28-6100-2, Section 13, "Control Card Formats," and to the reference manual, IBM 7090/7094 Operating Systems: Basic Monitor (IBSYS), form C28-6248.

Only designations for 729 tapes may be used for 7090 9PAC files. Designations referring to other I/O units may only be used for files referenced in hand calculations.

In disk unit assignment for files referenced in hand calculations, only this field is used. No symbolic channel assignments are allowed. The symbol XDAM represents the only assignment available, where X is any real channel A-H, D is disk designation, A is access mechanism 0 or 1, and M is module 0 through 9.

● p.25 Replace the entries for columns 22-25 with:

The unit assignment routines used by 7090 9PAC are identical with those used by 709/7090 IOCS.

If a disk designation appears in columns 18-21, this field must contain a numeric value indicating the number of consecutive cylinders allotted to IOCS for this file. The number cannot exceed 250.

p.26 The description for entries in column 36, Restart Flag, should have the following added:

N	Suppress repositioning when restarting.
b	Reposition when restarting.

● p.26 Replace the entry for column 37 with:

Write-Check option flag (disk only).

Blank-No write checking

W -Write checking is performed for partial and total block output files.

● p.26 Replace the entries for columns 38-41 with:

If labeled tape files, this field must contain the sequence number of the first reel. This is designated as the "load point" for the file or BOT. If a disk designation appears in columns 18-21, this field must contain the starting cylinder number (0-249) on the module where sequential processing by IOCS begins. From columns 22-25 and this field, IOCS will compute the valid binary head and track limits for the file.

p.26 Replace the entry for columns 42-43 with the following:

42	Preprocessing Input Density	For File Number 09, must contain one of the following codes to indicate the density of input to the horizontal preprocessing phase. H High L Low Must be blank for all other files.
43	Preprocessing Unit	For File Number 09, must contain one of the following codes to indicate the location of input to the horizontal preprocessing phase. * SYSIN1 b Separate tape

● p.26 Replace the entry for columns 44-48 with:

If labeled tape files, this field may contain a file serial number to identify the file. If a disk designation appears in columns 18-21, columns 44-45 may contain the HA2 identifier. This means that IBEDT has previously written this identical HA2 on those cylinders which are designated by columns 22-25 and columns 38-41. IBEDT must also have written the format track.

p.29 Column 7 of B record will always be blank.

ADDENDA AND ERRATA TO PART 2

The following addenda and corrections should be made on the pages listed.

p. 5 Replace the last sentence of the first paragraph under "Horizontal Change" with the following:

This preprocessing becomes the first phase of a two-phase object program. The input is an unblocked BCD file of card images. The output is a binary file (see Appendix B) which is input to the File Processor as a Change Data Input File with a *FILE card for File Number 09. The file does not have an attached dictionary nor may it be labeled or have check sums specified. All other physical characteristics are variable except block size which is determined internally. Also, columns 42 and 43 are used to specify the density and location of input to the preprocessing phase.

p.27 The last sentence of the description for column 47, Replace Only Code, should be replaced with the following:

If blank, a 1 is assumed for BCD fields and a 0 is assumed for BIN fields.

● p.40 Under columns 17-20, Contents should be changed to read: Change Report Field Number or Field Name

● p.40 The first sentence of the Description under columns 17-20 should be changed to read:

Must contain a literal field designation (numeric or alphabetic) by which the field being reported may later be referenced.

p.40 The Literal Field Number in the Description for columns 17-20, specifying the description of field changed, should be DESC, not DSC.

p.45 The description of the PUT macro-instruction should be replaced with the following:

The PUT macro-instruction places the contents of the BCD or binary buffer in the field referenced by its address portion, depending on whether the field is BCD or binary, respectively. The PUT macro-instruction will also check the dictionary for the field and, if the field is to be summed, it will automatically be added to the appropriate summary fields.

PUT may refer only to the current master file record type. PUT should not be used in pre-calculations with I action since the skeleton has not yet been inserted.

p.78 Add the following error description:

481X EOF not expected on change and error file.

ADDENDA AND ERRATA TO PART 3

The following addenda and corrections should be made on the pages listed.

p. 4 The last sentence of the first paragraph which begins "A detail-line...," should be replaced with:

Common usage is to make the carriage control character a one (1) and thus convert a detail-line header to a page header and lines per heading to lines per page.

● p. 6 The end of the last sentence of the first paragraph following item four: "...an error message will be written during generation." should be changed to:

the high-order characters will be truncated and an error message will be written during generation.

p. 8 Add to the first paragraph of "Format of Output Records" the following:

The record length is computed to be the sum of the field lengths. However, a greater length may be specified—in which case, padding is added to the end of the record.

p. 9 Add as a second paragraph under "Creation of the Output Dictionary" the following:

A single output file may have various record types created in more than one packet. Each packet must contain dictionary definitions for only those record types being created. When one record type (of one file) is created in more than one packet, the dictionary definitions must be carried in each packet and must be identical.

p.14 The last complete sentence on the page should have the word "detail" added as the next-to-last word. It should read:

A second action which can be controlled by the tests is the printing of a detail line.

p.16 Insert the following paragraph immediately before the heading "Emitting Data":

EXAMPLE 3

The report illustrated in Example 1, page 15, could be printed using an "or" condition rather than an "and" condition. The coding for this example is listed below.

	Func-										
Rec-	tion or	Value Field	Classification								
ord	Input	or Field	Table Columns								Format
Type	Field	Name	0	1	2	3	4	5	6	7	Field
10	0003	SALARY									D10026
10	CFYH	10000-14999					2				
10	CFYH	15001-20000					2				
10	ACT						2				

These lines cause acceptance of all salaries in the ranges $10,000 to $14,999 or $15,001 to $20,000.

p.22 Column 12 used only in connection with I-language. The I-string output feature will not be implemented.

● p.23 Change the description under column 29 to read:

May contain S, D, or P to indicate to the operator how the printer is to be set. S represents Single space; D represents Double space; P stands for Program control. Program control indicates that printer operation depends on the carriage control character in column 13 of the Report Format.

● p.24 The following should be added to the description under column 13:

(P stands for Program control.)

● p.25 The first part of the first sentence of the Description under columns 7-10 should be changed to read:

May contain either a function or the field number (numeric or alphabetic) of the field to be operated upon;

p.26 The description for column 47, Control Break Level, should begin as follows:

If it desired to cause a control break on a change in the field specified in columns 5-10 (or a portion of the field as specified in columns 30-35), this column may contain. . . .

p.29 The File No. and Rec. fields (columns 7-9) of the A01 line should not be shaded.

p.30 Add Record Length as columns 12-15 of the illustration of Form X28-6148.

p.30 Replace the entry for columns 12-17 with the following:

12-15	Record Length	May be blank if record length is to equal to sum of field lengths. If a larger number is entered, the record is padded to this length. (This is useful when the records are to be sorted with a fixed-length sort program.)
16-17		Must be blank.

p.36 The entry in the column "May Insert Be Conditional?" for Insert Point Symbol Hx should be No.

● p.36 The last symbol on the page should be changed to #yBr.

p.37 Symbol #$EOr should be changed to #$E0r.

● p.37 The last symbol on the page should be ##WRr where r is report number (see top of page 36).

p.38 Add the following to the table of Reference Points:

Symbol	Description
ZERO	This has the value 0.
ONE	This has the value 1.
TWO	This has the value 2.
THREE	This has the value 3.
FOUR	This has the value 4.
FIVE	This has the value 5.
SIX	This has the value 6.
SEVEN	This has the value 7.
EIGHT	This has the value 8.
NINE	This has the value 9.
TEN	This has the value 10.

● p.39 The second sentence under The CALC Pseudo-Instruction should begin:

Columns 8-11 of the first card. . . .

p.40 Replace Item 1 with the following:

Each instruction defining an associated storage location must have a unique location symbol of the form:

C x

where x is a four-digit number.

p.40 Replace rule 2 with the following:

2. Each hand calculation is limited to 100 "C" symbols.

p.40 Delete rule 3.

p.41 Add to Item 1 of the GET macro-instruction:

and will leave the value right-justified in the AC if the field is binary or a BCD field of six or less characters.

p.41 The last paragraph of the GET macro-instruction should be replaced with:

GET may destroy the contents of the MQ.

p.43 Insert the following subroutine:

BCDCHS

This subroutine may be used to change the sign of a BCD number.

1. Store the least significant six characters, right-justified, in BCDBUF.
2. Execute the following calling sequence:

```
        TSX    BCDCHS,4
```

3. The BCD number, with sign changed, will be located in BCDBUF.

● p.43 Replace the first paragraph of "Sequence of Hand Calculations" with:

The insert point for a Reports Generator hand calculation is determined by the following (in the order listed):

1. Associated record type.
2. Format line number of symbolic insert point.
3. Input record type (if applicable).
4. Classification table usage (if applicable).

If more than one hand calculation is to be performed at the same insert point, it will be executed in the order of the occurrence of the CAL parameter cards.

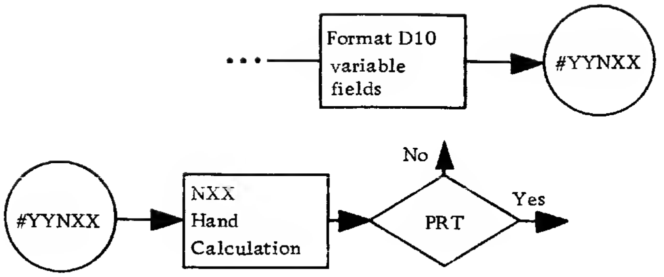
p.47 The block of coding for "Write subroutine for packet 01" and for "Write subroutine for packet 02" should be the last entry associated with each packet rather than stacked at the end of all packets.

● p.48 Delete unconditional system message 6.

p.48 Unconditional system message 7 should read:

FILE NAME XXXXXX LOGICAL RECORDS

● p.54 Include insert point for NXX hand calculation in detail line flow charts between #YYNXX and the PRT test.



● The mnemonics CFNH, CFNL, and RCFL should be changed to CYNH, CYNL, and RCYL, respectively. They occur in the text as follows:

CFNH to CYNH: page 13, last paragraph, line 1; page 15, under <u>EXAMPLE 1,</u> line 3 of the first chart; page 16, line 8 of the first chart; in the Index on page 61, under "Classify Not High, CFNH."

CFNL to CYNL: page 14, the paragraph preceding "Functions Used to Specify Action," line 4; page 15, under <u>EXAMPLE 1,</u> line 3 of the second chart; page 25, under "Field Number or Function," line 4; page 33, under "Field Number or Function," line 4; in the Index on page 61, under "Classify Not Low, CFNL."

RCFL to RCYL: page 14, the paragraph preceding "Functions Used to Specify Action," line 4; in the Index on page 61, under "Reclassify Low, RCFL."

ADDENDA AND ERRATA TO PART 4

The following addenda and corrections should be made on the pages listed.

p. 1 Replace paragraph 1 of the "Introduction" with:

7090 9PAC operates under the control of the IBSYS Basic Monitor which is described in the reference manual, IBM 7090/7094 Operating Systems: Basic Monitor (IBSYS), form C28-6248.

p. 1 Replace the first line of paragraph 5 of the "Introduction" with:

7090 9PAC uses 7090 IOCS, Version C with 1301 Sequential Capabilities as described in IBM 709/7090 Input/Output Control System reference manual, form C28-6100-2 and in IBM 709/7090 Input/Output Control System: Supplement for 1301 Sequential Capabilities bulletin, form J28-6204-1. It has been modified by 7090 9PAC as follows:

p. 1 Add the following to the note under "Uses of IBSYS Units" (paragraph 2):

SYSLB may be either a 729 tape or 1301 disk, whereas all other IBSYS functions must be 729 tapes.

p. 2 Replace line 7 with the following:

SYSUT1 Used only by RG if SYSOU1 is designated as an output unit and is the only output unit

p. 4 Replace the second line from the bottom of the page with:

9PAC 13 RG fixed object routines, e.g., GET, SET, counts

PREPARATION OF 9PAC FILES FOR USE WITH 7090/7094 SORT (729 - FIXED LENGTH)

- The 7090/7094 Sort (729 - Fixed Length) under the Basic Monitor will sort fixed-length records with control fields in the same respective locations in each record. In order to obtain output files from 9PAC which may be used as input to this sort program, it is necessary to make certain special provisions. If a 9PAC file containing a dictionary is used as input to Sort, the sort program requires the following:

(1) the first reel of the file must be the first mounted, since Sort checks only this reel for a dictionary; and (2) the dictionary must be in the same density as the label (if no label is specified, Sort assumes that the dictionary is in high density). The sort program will copy the dictionary onto the first reel of its final output file. The sort program is described in the bulletin, IBM 7090/7094 Generalized Sorting Program: 7090/7094 Sort (729 - Fixed Length), form J28-6217.

PREPARATION OF FILE PROCESSOR FILES

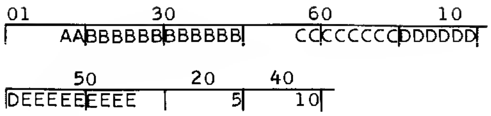
1. A single record type may be specified for the output file, in which case all records will necessarily be of the same length, with control fields in corresponding locations.

2. If two or more record types are specified, the format in which the File Processor arranges output records must be considered. The fields are grouped, with all unpacked BCD first, all packed BCD second, and all binary last. Fields within each group are in order of ascending field numbers.

For example, assume that a user wishes to specify two record types, numbered 10 and 20, respectively. Within record type 10, there are seven fields:

Field Number	Length (in characters)	Mode
01	2	100
10	7	110
20	5	001
30	12	100
40	10	001
50	9	110
60	8	100

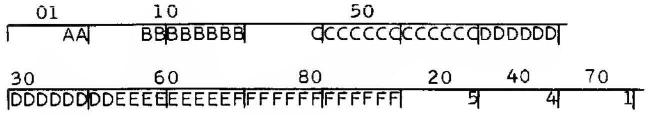
The File Processor will arrange these fields in the following format on the output file:



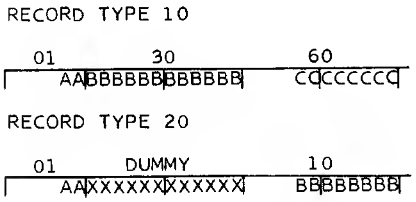
Within record type 20, there are nine fields:

Field Number	Length (in characters)	Mode
01	2	100
10	8	100
20	5	001
30	14	110
40	4	001
50	13	100
60	9	110
70	1	001
80	13	110

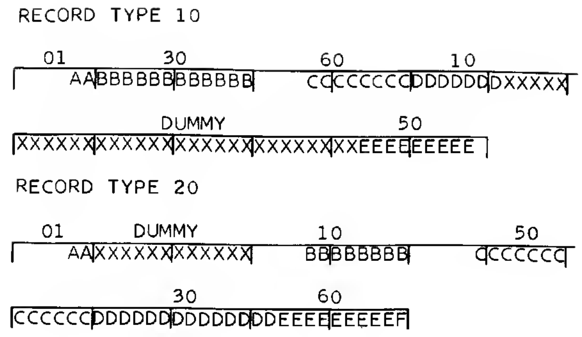
The File Processor will arrange these fields in the following format on the output file:



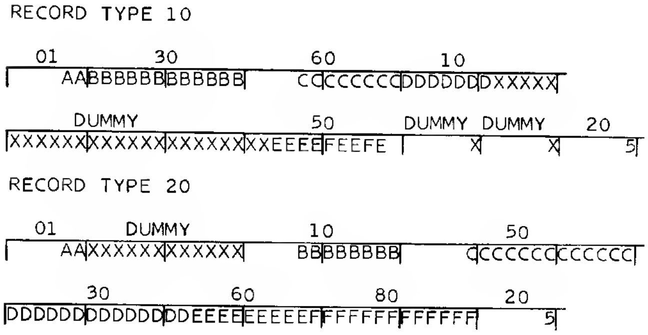
If the user wishes to sort the output file on fields 60, 50, and 20 of record type 10, which correspond to fields 10, 60, and 20 of record type 20, respectively, he must insert certain dummy fields to insure correct positioning of these control fields. A dummy field with a number 2-9, 7 to 12 characters in length, in unpacked BCD, inserted in record type 20, will cause fields 60 and 10 to be in the same position for all records.



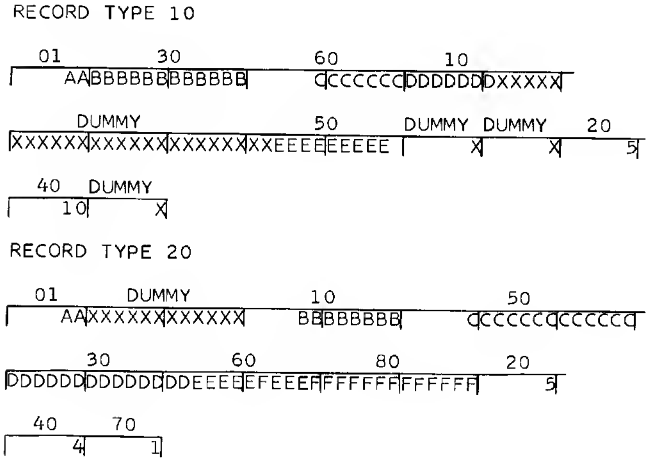
A dummy field with a number 11-49, 25 characters in length, in packed BCD, inserted in record type 10, will cause fields 50 and 60 to be in the same position:



Two dummy fields, numbered lower than 20 and in the binary mode, inserted in record type 10, will cause fields 20 and 20 to be in the same position.



Finally, a dummy field numbered higher than 40 and in the binary mode, inserted in record type 10, will cause all records to be of equal length.



PREPARATION OF REPORTS GENERATOR FILES

1. As with the File Processor, a single record type may be specified for a record output file, thus insuring fixed-length records with control fields in corresponding locations.

2. If two or more record types are specified, two options are available:
   a. The user may specify field increment in the F-record lines of the Dictionary Definition form for all fields in all record types. To obtain proper positioning, field increments for corresponding control fields must be the same in all record types. If record length is indicated in columns 12-15 of the E-record line, it must be constant for all record types; the Reports Generator will fill out shorter records to the desired length. If record length is not indicated, the user must add the correct number of dummy fields necessary to make all record types equal in length.
   b. If field increment is not specified, the user must determine the positions of his control fields and set the length of his records equal in the same manner as described under "Preparation of File Processor Files," with the following exception: if record length is indicated in columns 12-15 of the E-record line, the Reports Generator will automatically fill out records to the desired length, which must be constant for all record types.

3. Although record output will normally be used in creating files for sorting, report output may also be used. A record type field is not necessary for use with the fixed-length sort. Corresponding control fields must be in the same position in all print lines. All print lines for a given report (or set of reports stacked on a single tape unit) are automatically given a constant length. This length is equal to that of the longest line being written for that report (or tape unit). The record length, in 7090 words, may be taken directly from the Report Definition Format form, with the following provision: print lines of 19 words or less are given an additional word of blanks to allow for a minus sign (in the case of grouped output, this word also contains the group mark).

FILE PROCESSOR WITH REPORTS GENERATOR CAPABILITIES

INTRODUCTION

For large applications involving many updating functions or requiring many different reports, memory size limitations may require that File Processor and Reports Generator be used as separate programs. If this is the case, the inter-job and/or intersystem tape assignment routines incorporated in 7090 9PAC provide an efficient mode of operation. These are described in: IBM 7090/7094 Operating Systems: Basic Monitor (IBSYS) reference manual, form C28-6248.

For smaller applications requiring few updating functions and reports, it may be more efficient to update (or create) the file and simultaneously extract reports and/or records in the same pass of the master file. This can be done by writing the parameters independently and later combining the parameter decks in the manner described below. This method provides for parallel debugging of the Reports Generator and File Processor phases. It also allows later segmentation if core size is exceeded as more report and/or updating functions are added.

GENERATION OF OBJECT PROGRAMS

In an FPGEN job, after all cards have been read and if Reports Generator input (file #10) is present, the FPRGSW is set ON.

When FPRGSW is set ON, scratch files for both the File Processor and the Reports Generator are assigned as usual. The File Processor is entered and all instructions are generated. Before missing addresses are assigned and the final assembly made, the FPRGSW switch is tested. If ON, the Change and Error file dictionary is copied behind the Master Output dictionary. Provisions are made for the Reports Generator to use this "combined" dictionary to describe its input, and control passes to the Reports Generator.

The Reports Generator reads its parameter cards and completely generates its object program. Before execution, the FPRGSW is tested and if ON, control is returned to the File Processor where available core is divided into I/O buffers, missing addresses are assigned, the final assembly is made, and the object program is executed.

OBJECT PROGRAMS

Both the File Processor and the Reports Generator operate using common buffering routines to locate a logical record within an input block buffer or to locate available space for a logical record within an output block buffer. Index registers 1 and 2 and certain key tables are used for communication between the buffering routines (fixed program contained in record 9PAC 14) and the generated object programs. Tape reading and writing occurs whenever the buffering routines (9PAC 14) encounter and end of buffer block. The buffering routines also set aside those records which must be saved for later use by either the File Processor or the Reports Generator program.

The above logic permits object programs of the File Processor and Reports Generator to be combined in the following flow:

1. The File Processor enters 9PAC 14 to locate input records.
2. 9PAC 14 returns to the File Processor for processing.
3. The File Processor formats an output record in the buffer (either Master file or Change and Error Report file) and enters 9PAC 14 to locate space for the next output record.
4. 9PAC 14 records the entry point from the File Processor and transfers control to the Reports Generator giving the File Processor output record location as Reports Generator input record location. Note that the Reports Generator will receive records from both the Master file and Change and Error Report file in a single pass.
5. The Reports Generator formats a report line and/or record in the buffer and enters 9PAC 14 for the location of the next available output space.
6. 9PAC 14 returns to the Reports Generator.
7. After the Reports Generator has formatted all report lines (or records) and accumulated all information from the input record, it enters 9PAC 14 expecting to receive the next input record.
8. However, 9PAC 14 returns to the File Processor giving the location of the next output record space requested in step 3.

It should be noted that the order in which Master file records and Change and Error file records are received by the Reports Generator is determined by the order in which they are produced by the File Processor. Records from the two files would normally be processed by separate Reports Generator packets. Minus and plus zoning over the leading record type character distinguish Change and Error records respectively.

PARAMETER DECKS

The *JOB card must specify FPGEN as the job type. The *JOB card entries: Job Name, Punch Indicator, and Switches 1, 2, and 3, apply to both programs. Other entries apply only to the File Processor section. (Note that a binary deck may be requested and subsequent runs made with an FPBIN *JOB card).

The *FILE cards for both File Processor and Reports Generator files follow the *JOB card. The *FILE card for file #10 (Reports Generator input description) should be identical to the *FILE card for file #7 (File Processor master output) except for file type which is indicated as input. (It should be noted that the presence of an *FILE card for file #10 is the signal to the File Processor that Reports Generator parameters will also be present.)

An *END card should separate the File Processor parameters from the Reports Generator parameters.

A generation deck with File Processor and Reports Generator combined would thus have the following format:

```
*JOB      FPGEN       ⎫
*FILE                 ⎪  For File Processor
  .                   ⎬  and Reports Generator
  .                   ⎪  with no duplicate file
  .                   ⎭  numbers
*FILE
*END
File Processor
  parameters
  .
  .
  .
*END
Reports Generator
  parameters
  .
  .
  .
*END                      ⎫
File Processor hori-      ⎬  Optional
  zontal input date       ⎭
EOF (tape mark)           ⎫
File Processor            ⎬  Optional
  update file             ⎭
EOF (tape mark)
```

A binary deck would have the format listed below. Items preceded by P would have been punched when the generation deck was processed.

```
      *JOB     FPBIN
P     *FILE
        .
        .
        .
P     *FILE
P     *END
P     File Processor output
      dictionaries
P     Reports Generator
      output dictionaries
P     *END
P     File Processor Hori-       ⎫
      zontal Preprocessing       ⎪
      program                    ⎪
        .                        ⎬  Punched only
        .                        ⎪  if needed
        .                        ⎪
P     TRA to execute Pre-        ⎪
      processing program         ⎭
P     *END
      Horizontal input date      ⎫  Optional
      EOF (tape mark)            ⎭
P     Binary object program
      cards
        .
        .
        .
P     TRA to execute object
      program
P     *END
      File Processor update         Optional
      file
      EOF (tape mark)
```

PUBLICATIONS

Following is a list of IBM publications which may be of interest to the reader:

Form Number	Title
REFERENCE MANUALS	
A22-6528-4	IBM 7090 Data Processing System
A22-6536	IBM 709 Data Processing System
C28-6100-2	IBM 709/7090 Input/Output Control System
C28-6248	IBM 7090/7094 Operating Systems: Basic Monitor (IBSYS)
GENERAL INFORMATION MANUALS	
D22-6508-2	IBM 709/7090 Data Processing System
D22-6576-2	IBM 1301 Disk Storage with IBM 7000 Series Data Processing Systems
F28-8043	IBM Commercial Translator
F28-8053-1	COBOL
F28-8074-1	FORTRAN
BULLETINS	
G22-6505-1	IBM 7090 Data Processing System
G22-6595-2	IBM 1301 Disk Storage
J28-6080	IBM 709 Utility Programs
J28-6166	SHARE 7090 9PAC - Part 1: Introduction and General Principles
J28-6167	SHARE 7090 9PAC - Part 2: The File Processor
J28-6168	SHARE 7090 9PAC - Part 3: The Reports Generator
J28-6215	SHARE 7090 9PAC - Part 4: Operating Instructions
J28-6169-1	IBM 709/7090 Commercial Translator Processor
J28-6184	IBM 7094 Programs and Programming Systems
J28-6204-1	IBM 709/7090 Input/Output Control System: Preliminary Specifications for 1301 Sequential Capabilities
J28-6217	IBM 7090/7094 Generalized Sorting Program: 7090/7094 Sort (729 - Fixed Length)
J28-6223-1	IBM 7090/7094 Utility Routines for IBM 1301 Disk Storage
J28-6258	709/7090 Commercial Translator Processor - Supplement for 1301 Sequential Operations
J28-8072	Addenda to the Commercial Translator Manual
SOS REFERENCE MANUAL	
	SOS Reference Manual, consisting of loose-leaf binder (Form X28-1213) and the following distributions: (1) 328-1219; (2) 328-1262; (3) 328-1377; (4) 328-1395; (5) 328-1406.

IBM
International Business Machines Corporation
Data Processing Division, 112 East Post Road, White Plains, N. Y.